

200

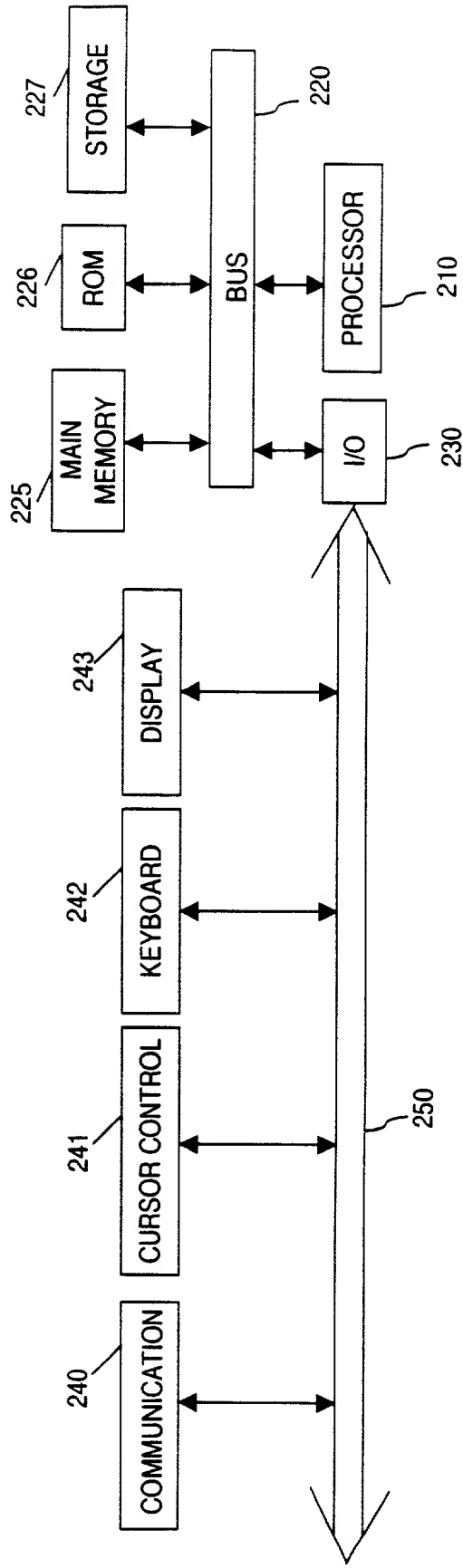


FIG. 1

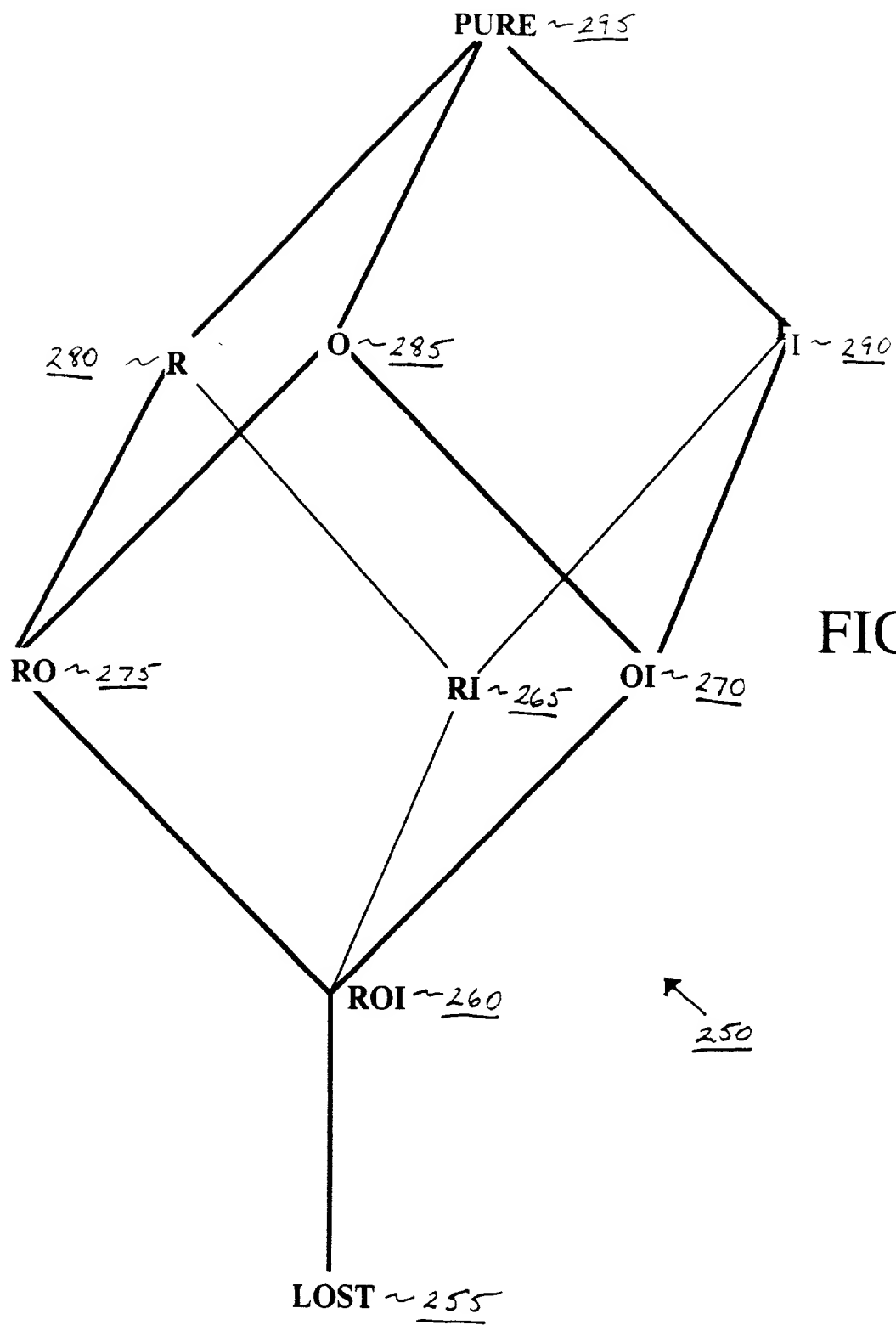


FIG. 2

250

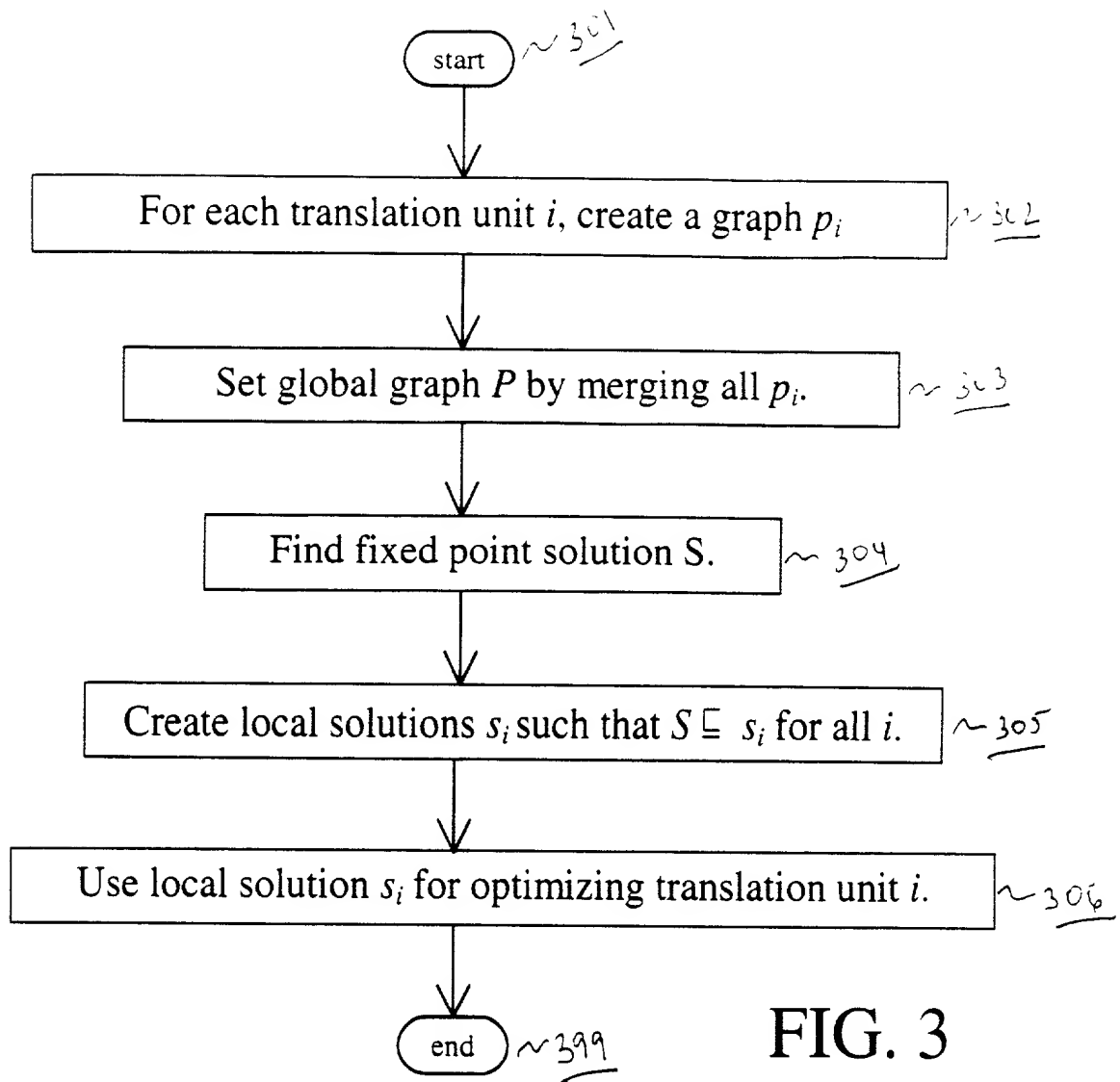
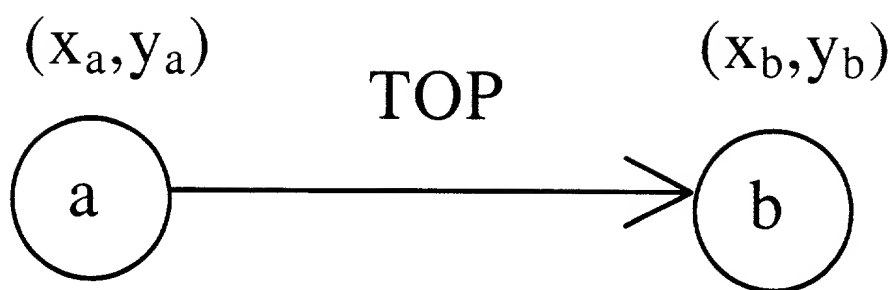


FIG. 3

# FIG. 4A

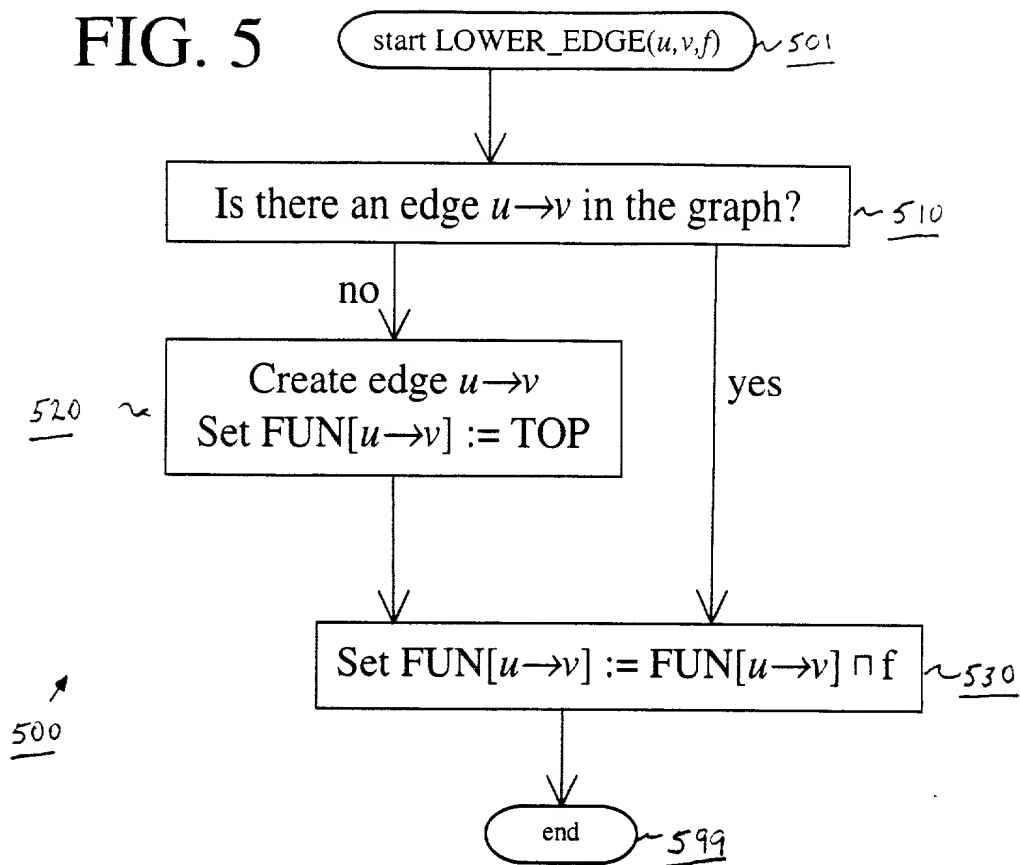
Function	Function (x,y)
<u>410</u> ~ TOP	(PURE,PURE)
<u>420</u> ~ COPY	(y,y)
<u>430</u> ~ IN_TO_LOST	if $y \leq I \Rightarrow (LOST,LOST)$ otherwise $\Rightarrow (PURE,PURE)$
<u>440</u> ~ UNRETURN	if $y = LOST \Rightarrow (LOST,LOST)$ otherwise $\Rightarrow (z,z)$ where $z = y \sqcup OI$
<u>450</u> ~ COPY_AND_IN_TO_LOST	if $y \leq I \Rightarrow (LOST,LOST)$ otherwise $\Rightarrow (y,y)$
<u>460</u> ~ CAT_FORMAL	(y,PURE)
<u>470</u> ~ CAT_ACTUAL	(PURE,y)
<u>480</u> ~ GATE	if $x = LOST \Rightarrow (LOST,LOST)$ else if $x \leq R \Rightarrow (z,z)$ where $z = (x \sqcup OI) \sqcap y$ else $(z,z)$ where $z = (x \sqcup OI)$

FIG. 4A



**FIG. 4B**

FIG. 5



600 ↘

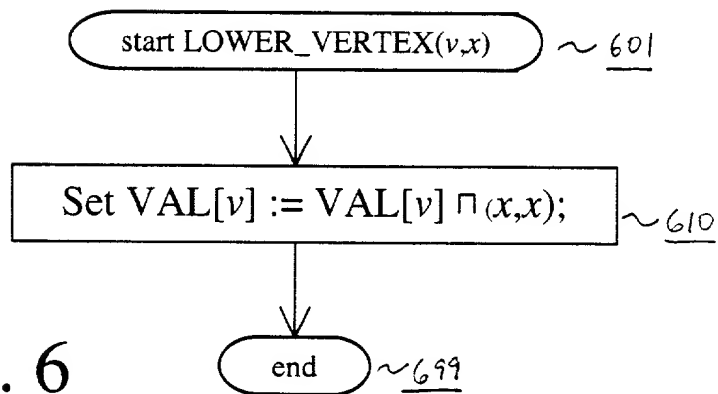
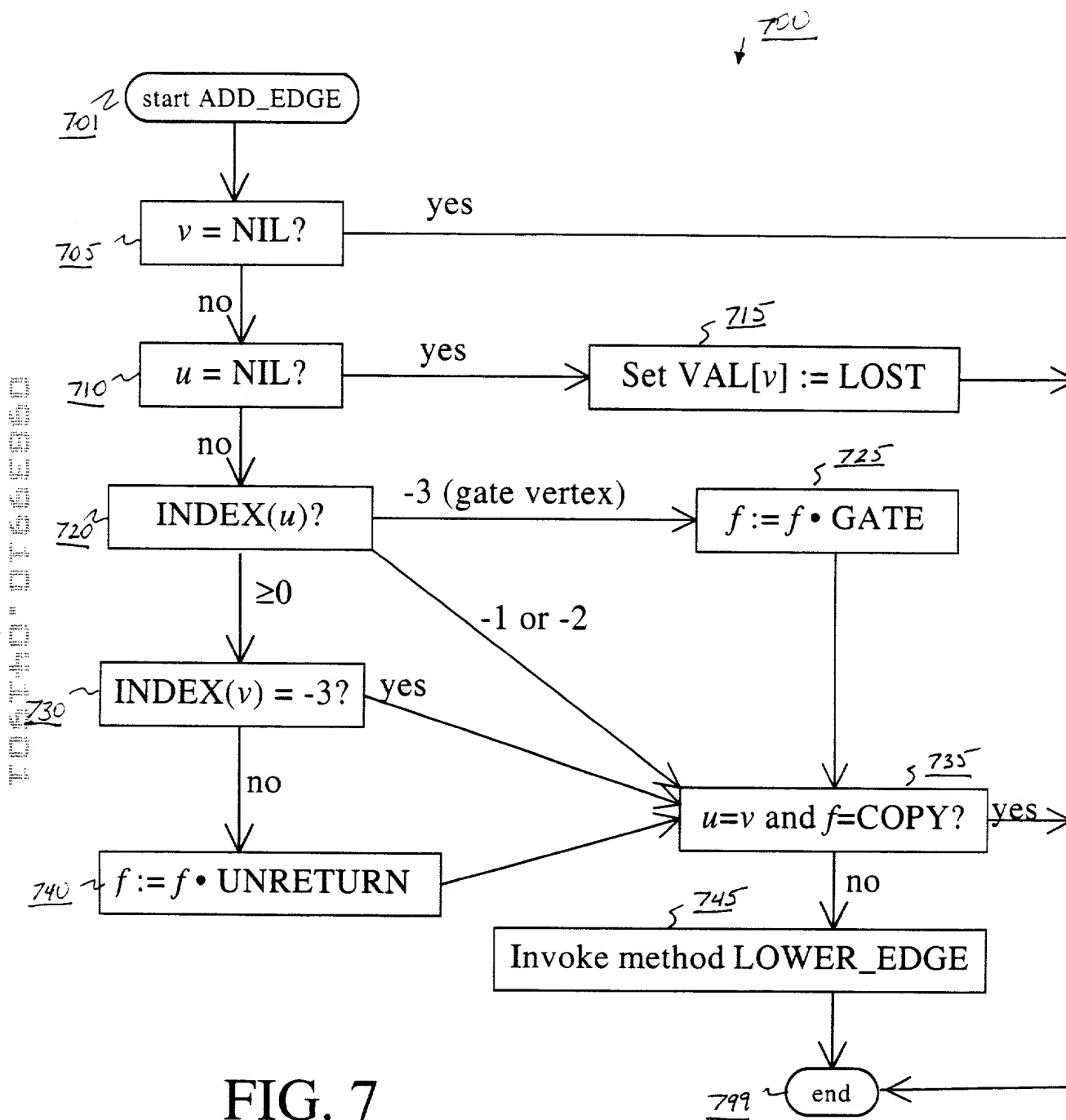
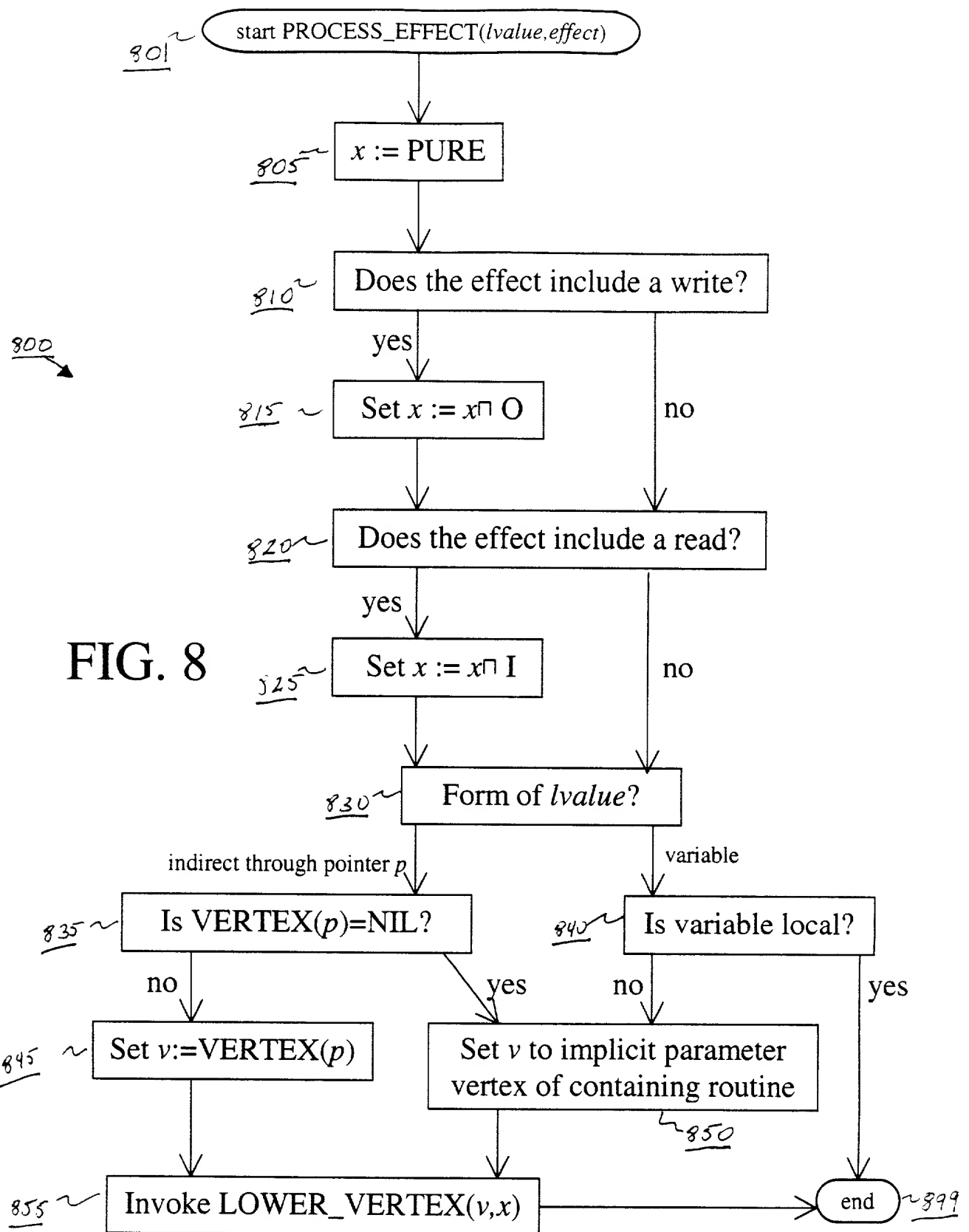


FIG. 6





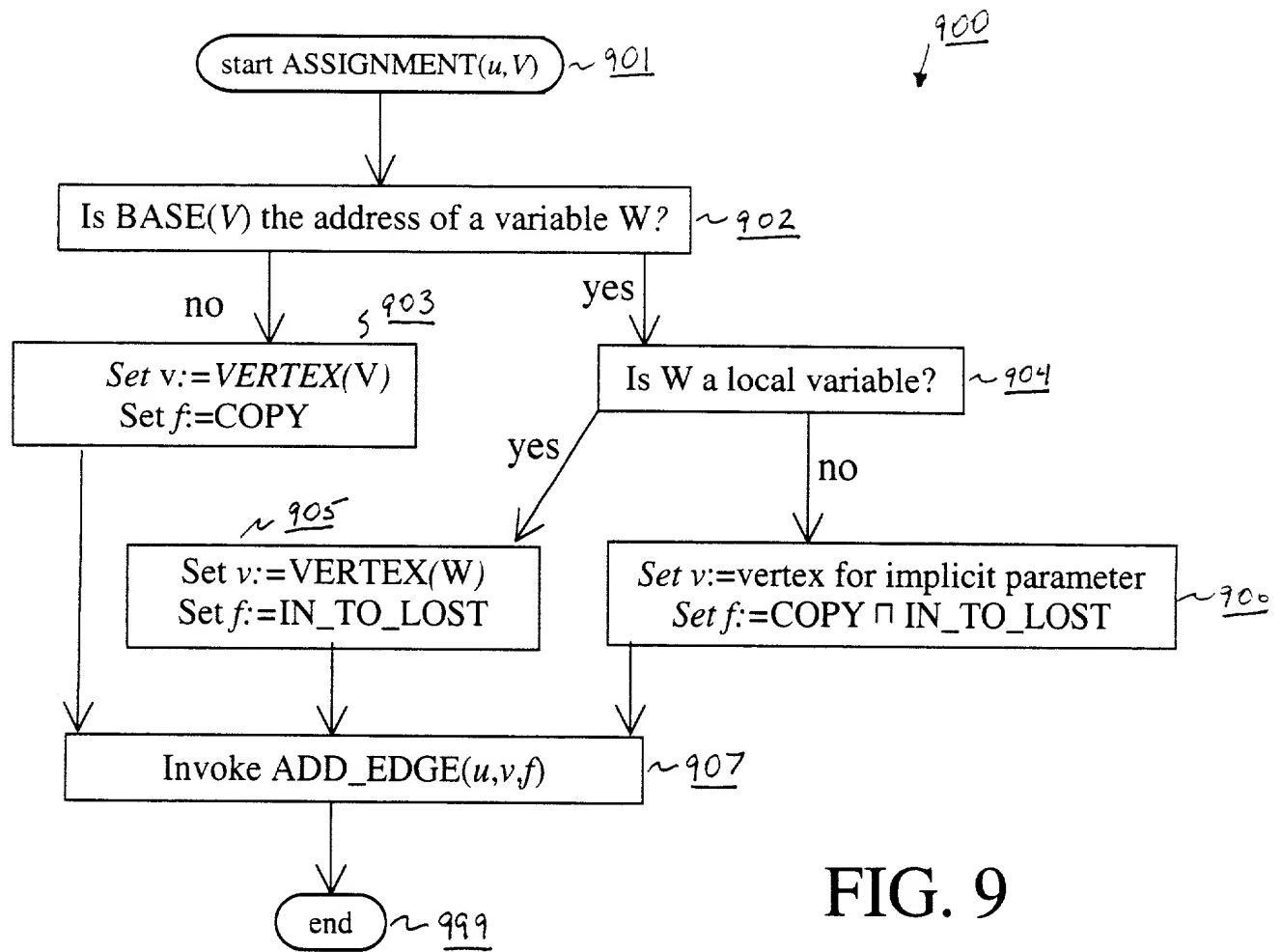


FIG. 9

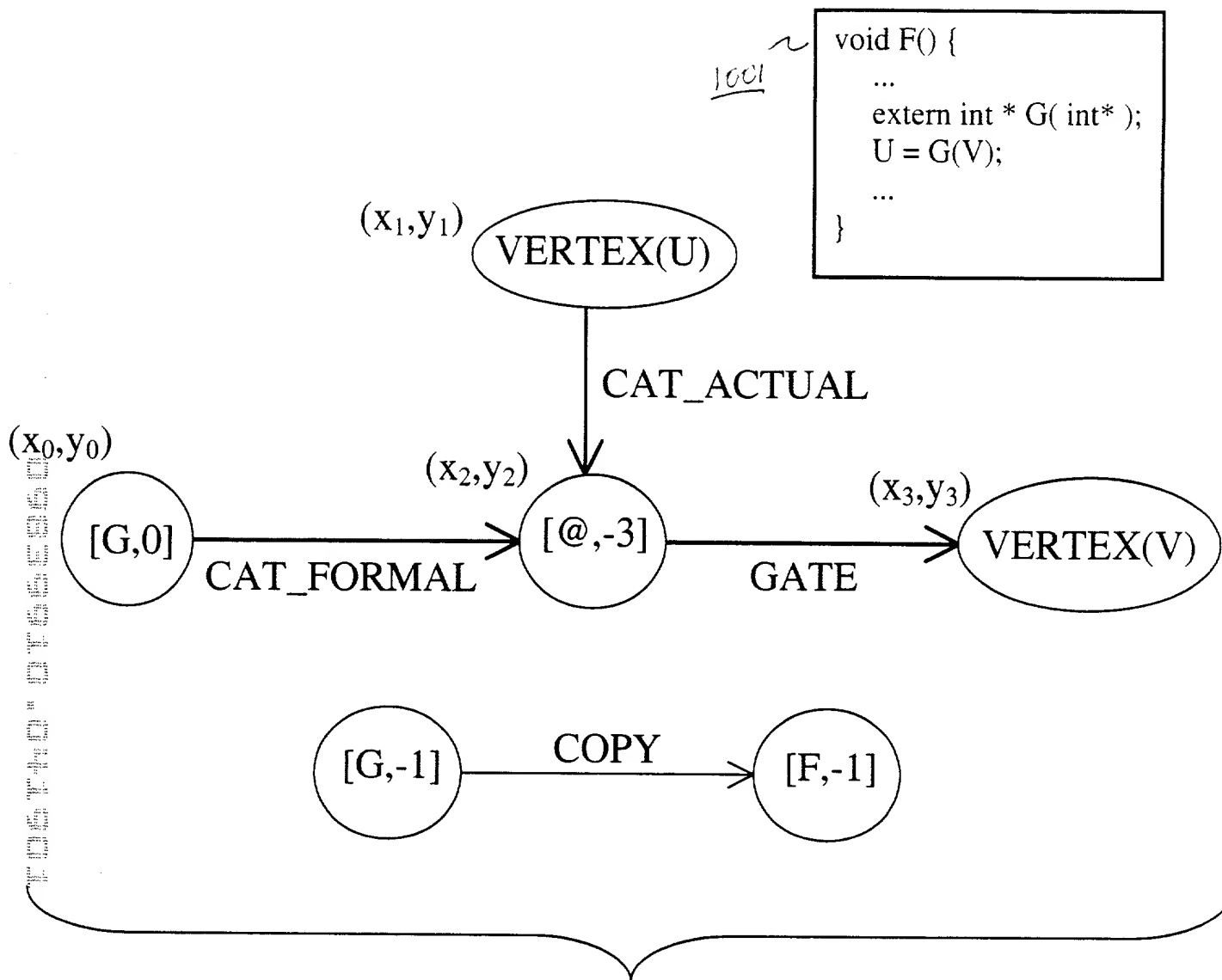


FIG. 10

// Translation unit #1

```
int* f( int* a, int* b, int n ) {  
    int *c = a;  
    if( n>0 ) {  
        int* d = a+1;  
        int* e = b+1;  
        int* z = f( d, e, n-1 );  
        c = z-1;  
        *c = *b;  
    }  
    return c;  
}
```

// Translation unit #2

```
extern int* f(int* a, int* b, int n );  
  
void g( int* p ) {  
    int y[10];  
    f( &y[0], p, 10 );  
}
```

FIG. 11

